GOVERNMENT COLLEGE OF ENGINEERING, CHHATRAPATI SAMBHAJINAGAR

(An autonomous Institute of Government of Maharashtra)

2024-25

Report on DBMS Project

On

"Medicare (University Healthcare Portal)"

Submitted By

Group 1

Himanshu Mishra	BT23F05F039
Saiesh Sharma	BT23F05F059

In partial fulfilment for the award of the degree

Of

Bachelor Of Engineering

In

Computer Science and Engineering

Under the Guidance of

Prof. Ashwini Sugave

GOVERNMENT COLLEGE OF ENGINEERING AURANAGABAD,

CHHATRAPATI SAMBHAJINAGAR

CERTIFICATE

This is to clarify that the project entitled “Medicare University Healthcare Portal” which is being submitted here for the award of Mini Project of “Bachelor of Engineering” in “Computer Science of Engineering” of “Government College of Engineering Aurangabad, Chhatrapati Sambhajinagar” affiliated to “Dr Babasaheb Ambedkar Marathwada University Chhatrapati Sambhajinagar”.

The Above project design of “FACLTY APPRAISAL SYSTEM” reports is submitted by:

Name:	**Roll number:**
Himanshu Mishra	BT23F05F039
Saiesh Sharma	BT23F05F039

Under the guidance of Prof. Ashwini Sugave about work is successfully completed and submitted by them.

Prof. Ashwini Sugave	**Prof. Sudhir Shikalpure**
Professor and mentor	Head of CSE Dept.

ABSTRACT

The University Healthcare Portal is a digital health solution developed to support university students, especially those living in hostels, by bridging the gap between students and medical professionals on campus. The platform offers a two-fold system: a web-based portal and a mobile application. Students can book dynamic appointments with doctors, chat in real-time using Socket.IO, receive prescriptions digitally, and order medicines directly through the platform. The system also incorporates real-time ambulance tracking and booking using Google Maps API and PubNub. The portal simplifies medical access, digitizes health records, and ensures swift emergency responses, improving campus healthcare infrastructure holistically.

TABLE OF CONTENT

SR NO.	CONTENT	Page No.
1.	ACKNOWLEDGEMENT	5
2.	INTRODUCTION	6
3.	Target Audience	7
4.	Objectives	8
5.	System Architecture	9
6.	Features Overview	11
7.	Implementation Details	14
8.	Tech Stack	16
9.	Future Scope and Conclusion	19
10.	References	20

ACKNOWLEDGEMENT

I am profoundly grateful to my esteemed Prof. Ashwini Sugave, whose expert guidance, constructive feedback, and continuous support were indispensable to the successful completion of this work. Their profound knowledge and insights have greatly enriched my understanding and have been instrumental in shaping the direction of this project.

We must acknowledge the staff of Data Center for their support and helpful guidance. We appreciate all the resources, references, and tools that have made this endeavor possible.

Thank you.

Himanshu Mishra BT23F05F039

Saiesh Sharma BT23F05F059

INTRODUCTION

Healthcare accessibility remains a pressing concern for students residing in university hostels, particularly in cases of minor health issues or emergencies. While these students are far from their homes and family doctors, they often lack convenient access to nearby hospitals or clinics. Scheduling appointments, waiting in long queues, and navigating transportation logistics can delay diagnosis and treatment, leading to neglected or exacerbated health conditions. These challenges are compounded during weekends, late-night hours, or examination seasons, when students are least inclined or able to seek medical help off-campus.

Recognizing these difficulties, our project introduces a centralized, digital healthcare solution designed specifically for the university campus environment. This University Healthcare Portal bridges the gap between students and medical professionals, allowing seamless access to healthcare services from the comfort of the hostel. Through our system, students can browse available doctors, book appointments based on dynamic 30-minute time slots, and receive timely treatment. The platform supports real-time communication through Socket.IO, enabling students to consult doctors instantly, discuss symptoms, and receive prescriptions without needing to visit the clinic physically.

Furthermore, the digital prescriptions are automatically added to a cart, allowing students to review, order, or purchase prescribed medicines online. In emergencies, students can instantly book an ambulance, share their live location with drivers, and notify emergency contacts. The integration of real-time maps using Google Maps API and PubNub ensures accurate ambulance tracking, allowing swift response and transparency for both drivers and patients.

This healthcare portal is accessible via a responsive, user-friendly web interface developed using React, and a mobile application built with React Native and Expo for on-the-go access. The system is powered by a robust backend built with Node.js and Express, and data is managed securely using MongoDB and Mongoose. The end-to-end system creates a comprehensive and efficient ecosystem for student healthcare, enabling timely treatment, preserving medical records, and reducing dependency on external facilities.

Target Audience

- **University Students**: The primary beneficiaries of this platform are students residing in university hostels, who often face difficulties in accessing timely medical care. The portal empowers them to independently book appointments, chat with doctors, receive prescriptions, and request emergency ambulance services. It also maintains a personal medical history that students can reference for ongoing care or referrals. The mobile app ensures that even students who are not near a computer can access care in urgent situations.

- **On-Campus Doctors**: For doctors, the portal provides an organized, digital environment to manage appointments, interact with students, and streamline recordkeeping. They can easily set their availability, review upcoming appointments, access patient history and symptoms, and prescribe medication directly through the chat interface. This system reduces paperwork, ensures efficient time utilization, and enhances the quality of care provided to students.

- **University Administration**: The platform aids university authorities in ensuring a responsive healthcare system for students. It allows for efficient management of medical services, oversight of doctor schedules, and tracking of emergency response times. With the potential future inclusion of an analytics dashboard, administrators can monitor student health trends, resource utilization, and service effectiveness.

- **Emergency Contacts**: In the event of a medical emergency, the system automatically notifies the student's emergency contacts—typically parents or guardians. This feature offers peace of mind to families and ensures that appropriate support networks are activated promptly during critical situations.

Objectives

- **Streamline Healthcare Access for Hostel Students**: The primary aim of the system is to ensure that students, particularly those staying in university hostels, can receive timely medical attention without the need to physically visit hospitals or clinics. By creating a centralized platform, we reduce delays and promote proactive health management among students.

- **Enable Seamless Appointment Booking with Real-Time Slot Updates**: The system facilitates dynamic booking of doctor appointments in 30-minute time slots. Once a slot is booked by a student, it becomes unavailable to others, ensuring efficiency and avoiding scheduling conflicts. Doctors can update their available days and timings at their convenience.

- **Implement Real-Time Chat and Prescription Sharing Using Socket.IO**: The platform incorporates a live chat system that allows students to communicate instantly with doctors, describe their symptoms, and receive timely responses. Prescriptions can be securely sent through the chat interface and are automatically added to the student's medicine cart for easy ordering.

- **Digitally Manage Student Medical Records**: Every interaction, prescription, and appointment is securely stored and made accessible to both students and doctors. This recordkeeping enhances continuity of care and provides valuable medical history for future consultations.

- **Develop a Robust Ambulance Tracking and Emergency Notification System**: In medical emergencies, students can quickly request an ambulance through the mobile app. The system shares real-time location between the ambulance driver and the patient using Google Maps API and PubNub. Simultaneously, it sends alerts to the student's emergency contacts, ensuring a rapid and coordinated response.

System Architecture

The architecture of the University Healthcare Portal is composed of a full-stack solution integrating web, mobile, and real-time communication technologies. Each component of the architecture plays a vital role in delivering a seamless and responsive experience for students, doctors, administrators, and ambulance operators.

- **Frontend (Web)**: The web portal is built using **React.js**, a powerful JavaScript library that enables the creation of dynamic and interactive user interfaces. It follows a component-based architecture that allows scalable development and efficient state management. **ShadCN UI** is used for prebuilt, accessible components which are customized for a clean and modern look. To enhance user onboarding, **React Joyride** is integrated to offer guided walkthroughs, helping new users understand the functionality of the portal step-by-step.
- **Frontend (Mobile)**: The mobile interface, created with **React Native** and powered by **Expo**, caters specifically to patients and ambulance drivers. The app supports real-time location tracking, push notifications, and two-way communication. Expo facilitates rapid prototyping and testing, ensuring smooth cross-platform performance on both Android and iOS devices.
- **Backend**: The server-side logic is implemented using **Node.js** and **Express.js**. Node.js offers non-blocking I/O and high performance, while Express simplifies API creation and routing. The backend handles tasks such as user authentication, chat management, appointment scheduling, and prescription processing.
- **Database**: The system uses **MongoDB**, a NoSQL database, to store structured and semi-structured data like user profiles, medical history, chat records, appointment slots, prescriptions, and ambulance booking logs. **Mongoose** is utilized for schema enforcement and querying, ensuring robust data modeling and validation.

- **Real-Time Services**:
    - **Socket.IO** is employed for real-time communication between students and doctors. It allows instant message exchange, presence detection, and message persistence, enhancing the responsiveness of consultations.
    - **PubNub** is used in the ambulance tracking module to support continuous location polling and updates between the driver and the patient, providing real-time visibility and geolocation synchronization.

- **APIs**: The portal integrates the **Google Maps API** to enable precise geolocation services for the ambulance tracking system. It allows rendering of live maps, route visualization, and direction tracking, ensuring efficient navigation and reduced emergency response times.

This architecture ensures that the platform remains highly responsive, scalable, and maintainable, with real-time capabilities and mobile-first responsiveness being core strengths.

Features Overview

The University Healthcare Portal offers a comprehensive suite of features to address healthcare needs for students and doctors, along with emergency support functionalities. The solution is designed with user-centric experiences in both web and mobile environments, ensuring accessibility, reliability, and real-time responsiveness.

1. Student Web Portal
The student portal serves as the primary interface for accessing healthcare services. Designed with ease of use in mind, it allows students to manage their medical appointments and emergencies from anywhere within the campus.

- **Login/Register Securely**: Secure authentication using JWT ensures only authorized users can access sensitive data. Bcrypt is used for password hashing to enhance security.

- **View Doctor Profiles and Book Appointments**: Students can browse available doctors, check their working hours, and book 30-minute slots dynamically. The system ensures real-time updates to slot availability to prevent double bookings.

- **Real-Time Chat with Doctors**: Implemented using Socket.IO, students can instantly connect with doctors, share symptoms, and receive consultation without visiting the clinic.

- **Receive and Manage Prescriptions**: Prescriptions shared during chat sessions are automatically stored in the student's account, with downloadable access for future reference.

- **Order Medicines via an Integrated Cart**: Medicines prescribed by doctors are added directly to a cart, allowing students to review and place orders seamlessly.

- **Book Ambulance and View Real-Time Location**: Students can request ambulance services and monitor the driver's real-time location through Google Maps integration.

- **Notify Emergency Contacts During Ambulance Requests**: Upon ambulance booking, automated alerts are sent via SMS to emergency contacts to ensure timely support.

Doctor Web Portal

The doctor portal is optimized for simplicity and effectiveness, allowing healthcare providers to manage their workflow efficiently and interact with students as needed.

- **Set Working Days and Times for Availability**: Doctors can log in and define their availability schedule using an intuitive calendar interface, which dynamically generates 30-minute appointment slots.

- **Accept/Decline Appointments**: Doctors can view upcoming appointments, confirm or reject bookings, and get notified about changes in their schedule.

- **Access Patient History, Symptoms, and Chat**: Prior to or during consultations, doctors can review the student's previous appointments, chat history, and submitted symptoms.

- **Prescribe Medication and Manage Records**: A built-in prescription editor allows doctors to send medication details directly to students, which are saved in the database and reflected in the student's medicine cart.

Mobile Application (Ambulance)

The mobile application was developed using React Native and Expo, offering tailored interfaces for both ambulance drivers and patients

- **Separate Portals for Ambulance Driver and Patient**: The driver app allows viewing of assigned bookings and navigation via GPS. The patient app allows booking and tracking of the ambulance.

- **Share and View Live Location Using Google Maps and PubNub**: Real-time geolocation is achieved through PubNub polling and displayed using Google Maps API, ensuring accurate tracking of ambulance movement.

- **Enable Text and Voice Notifications**: Voice and text alerts inform both parties about arrival times, current positions, and delays.

- **Book Ambulance with a One-Click Emergency Message System**: With a single tap, students can call for help and notify their guardians, streamlining the emergency response process.

4. Admin Panel (Future Scope)

To enhance scalability and oversight, a future implementation of the Admin Panel is proposed. It would provide administrators with comprehensive analytics and management capabilities.

- **Track Doctor Availability, Appointment Analytics, Emergency Calls**: Admins can monitor resource usage, peak demand hours, and service load to ensure optimal healthcare delivery.

- **Maintain Digital Health Records for Audit**: Securely store and manage historical health data for institutional audits and improvement strategies.

This broad feature set provides a robust and user-friendly system that adapts to the needs of all stakeholders in a university's healthcare ecosystem.

Implementation Details

The University Healthcare Portal's implementation focuses on real-time performance, data privacy, and service reliability to ensure a smooth experience for both patients and healthcare providers. Each technical feature is designed with end-user benefit in mind, backed by modern and scalable technologies:

JWT Authentication and Secure Password Hashing (bcrypt):
To enforce secure and persistent authentication, the portal uses JWT (JSON Web Tokens). Once a user logs in, a signed token is generated and stored on the client side (typically in HTTP-only cookies or localStorage). This token is required to access any protected route on the server. Bcrypt is used to hash passwords during registration and before every login attempt, ensuring that even if database credentials are compromised, plaintext passwords are never exposed.
Sample code:
js
CopyEdit
const token = jwt.sign({ id: user._id, role: user.role },
process.env.JWT_SECRET, { expiresIn: '1d' });

Real-Time Availability Updates for Appointment Slots:
Doctors configure their availability through a user-friendly dashboard. The backend automatically divides each available period into 30-minute time slots. When a student books a slot, that time is immediately marked unavailable, and this change is pushed in real-time to the frontend using socket events or REST-based polling, preventing double bookings.

Chat Stored in Real-Time and Asynchronously Using Socket.IO:
The chat module provides secure, real-time messaging between students and doctors. Every message is instantly transmitted using WebSockets and also stored in MongoDB for recordkeeping and offline viewing. The chat includes support for timestamps, typing indicators, and delivery status, offering a familiar experience similar to popular messaging platforms.

Key UX Benefit: Students can share symptoms instantly, attach text, and receive consultation even during late hours without needing to physically visit a clinic.

Doctor’s Prescribed Medicine is Parsed into a Digital Cart for Order:
Doctors prescribe medication during or after a chat session. This prescription is automatically parsed into a structured format — extracting medicine name, dosage, and duration. The system then populates the student’s cart, where they can choose to order directly or download the prescription for manual fulfillment. This also supports future integration with campus pharmacies.

PubNub Polling Used for Continuous Ambulance Tracking:
The ambulance module uses PubNub’s publish-subscribe architecture to provide real-time geolocation updates. As the ambulance driver’s device sends location coordinates at regular intervals, the patient’s interface receives and renders this movement on a Google Map, refreshing every few seconds. This builds transparency and helps students estimate arrival time.

Real-World Value: In medical emergencies, even a 5-minute delay in tracking can result in panic or miscommunication. Live updates reduce this uncertainty and help students stay calm.

Notification System with Twilio (Future Scope):

To add another layer of safety during emergencies, we propose integrating Twilio's messaging service. When a student books an ambulance, a preconfigured SMS is sent to their emergency contacts, including name, location, and a short description of the emergency. This allows family or guardians to stay informed and take additional action if needed.
Future Enhancements: These messages could be expanded to include a tracking link, Google Maps directions, or estimated arrival time of the ambulance.

Tech Stack

The University Healthcare Portal is built using a robust and modern technology stack, enabling seamless integration of web, mobile, real-time, and geolocation functionalities. Each layer of the stack has been carefully chosen to ensure optimal performance, maintainability, and scalability.

- **Frontend:**

React: Used to build the web portal with a modular, component-based architecture that ensures scalability and high performance. React's state management and virtual DOM make it ideal for dynamic interfaces like appointment scheduling and chat.

Tailwind CSS: A utility-first CSS framework used for rapidly styling components with custom design, ensuring responsiveness and consistency across devices.

ShadCN UI: Integrated for reusable and accessible UI components that are visually clean and easily customizable, helping accelerate frontend development while maintaining a professional look.

React Joyride: Utilized for onboarding and guided walkthroughs, especially helpful for first-time users to understand the layout and core functionalities like booking appointments or starting a chat.

React Native (Mobile): Powering the mobile app interface, React Native enables cross-platform development for both Android and iOS, allowing real-time tracking and chat to be available on the go.

- **Backend:**

Node.js: The core of the backend, chosen for its event-driven and non-blocking I/O model. It supports handling a large number of concurrent requests efficiently, making it ideal for real-time services.

Express: A minimalistic and flexible Node.js web framework used to structure REST APIs, route handling, middleware setup, and error management with ease.

- **Database:**

MongoDB with Mongoose: A NoSQL document-based database well-suited for storing varied data like user profiles, chats, prescriptions, and booking details. Mongoose provides schema enforcement and seamless object modeling.

- **Real-Time Technologies:**

Socket.IO: Enables bi-directional, low-latency communication for the real-time chat system. It handles message broadcasting, room joining, and user presence effectively.

PubNub: Used for the live tracking feature in the ambulance module. It offers real-time publish/subscribe infrastructure, ensuring that both driver and patient receive live updates.

- **Map Services:**

Google Maps API: Integrated for rendering maps, tracking live ambulance location, route visualization, and geolocation services. It provides detailed, accurate, and interactive maps that enhance emergency response.

This technology stack ensures high availability, scalability, and a responsive experience across all devices while laying a strong foundation for future enhancements and third-party integrations.

Challenges Faced

1. Ensuring Real-Time Synchronization of Dynamic Time Slots

Managing dynamic time slots for doctor appointments or ambulance availability in real-time posed a significant challenge. With multiple users accessing and modifying time slots simultaneously, we had to ensure that no conflicts occurred. This required implementing robust logic on both the frontend and backend to lock and update slots instantly, using WebSockets for immediate updates and fallback mechanisms to handle network delays or failures.

2. Managing Concurrent Chat Connections with Socket.IO

Implementing real-time chat using Socket.IO was complex due to the need for handling multiple concurrent connections efficiently. Each patient and medical staff member could initiate or join multiple chats, requiring us to build a scalable architecture that maintained persistent socket connections, handled room-based messaging, and ensured message delivery even during temporary disconnects or reconnections.

3. Securing Patient Health Data and Chat Logs

Given the sensitive nature of healthcare data, it was critical to implement strong security measures. We had to ensure end-to-end encryption for chat messages, secure authentication for accessing chat logs, and compliance with data protection regulations. This included implementing JWT-based auth, encrypting data at rest and in transit, and securing API endpoints against unauthorized access or injection attacks.

4. Synchronizing Ambulance Live Location Across Devices Using PubNub

Displaying live ambulance location in real-time to both patients and medical staff involved integrating PubNub's real-time data streaming. The challenge lay in maintaining low-latency, high-accuracy location updates without overloading the network or draining mobile device battery. We had to carefully optimize the frequency of GPS updates, handle intermittent connectivity, and sync location data consistently across all subscribed devices.

Future Scope

- **AI-based Diagnosis Support**: Based on symptoms and history
- **Video Consultation**: Integrate WebRTC or Twilio for calls
- **Pharmacy Integration**: API integration with campus/local pharmacies
- **Admin Analytics Dashboard**: Doctor workload, student health analytics
- **EMR Integration**: Support Electronic Medical Records compliance
- **Dark Mode & Accessibility Support**
- **PWA Version**: Progressive Web App to support offline access

Conclusion

The University Healthcare Portal is a complete ecosystem for healthcare on campus. From appointment booking to live ambulance tracking, it supports real-time, secure, and intelligent communication between students and medical professionals. With strong backend architecture, interactive frontend, and seamless integrations like Google Maps and Socket.IO, this system elevates student healthcare access, helping universities better manage health crises and routine consultations.

References

- Google Maps API Documentation
- PubNub API Docs
- Socket.IO Documentation
- MongoDB + Mongoose Docs
- React Native Ex